9th International Conference on Numerical Grid Generation in Computational Field Simulations, San Jose, California, June 11-18, 2005.

Advances in Software Tools for Pre-processing and Post-processing of Overset Grid Computations

NAS Technical Report, NAS-05-010
June 2005

William M. Chan

NASA Ames Research Center
Mail Stop T27B-1
Moffett Field, CA 94035-1000
wchan@nas.nasa.gov

Abstract

Recent developments in three pieces of software for performing pre-processing and post-processing work on numerical computations using overset grids are presented. The first is the OVERGRID graphical interface which provides a unified environment for the visualization, manipulation, generation and diagnostics of geometry and grids. Modules are also available for automatic boundary conditions detection, flow solver input preparation, multiple component dynamics animation and input preparation, simple solution viewing for moving components, and debris trajectory analysis input preparation. The second is the Chimera Grid Tools Script Library that enables rapid creation of grid generation scripts. A sample of recent applications will be described. The third is the OVERPLOT graphical interface for displaying and analyzing data history files generated by the flow solver. Data displayed include residuals, component forces and moments, number of supersonic and reverse flow points, and various dynamics parameters.

1. Introduction

A complex configuration modeled by Chimera overset structured grids [1] can be thought of as an unstructured collection of overlapping structured grids where neighboring grids overlap with comparable resolution. Grids are typically constructed to have one boundary conform to the body geometry of the object being modeled, while the other boundaries are allowed to float. The domain away from the body is often resolved using various levels of Cartesian grids that overlap with the near-body grids. This flexibility usually results in higher quality grids than the patched grid methods with abutting grid boundaries. With overset grid methods being used for increasingly complex flow simulation problems in the 1990's [2-11], a common complaint at the time was that there was a lack of software tools available that were specifically suited for overset structured grid generation. The Chimera Grid Tools (CGT) software project was then initiated to collect various existing tools such as hyperbolic grid generators [12] into one package, and to further develop new tools for efficient and easy-to-use overset grid generation, flow solver input creation, and flow solution post-processing. Other software packages with a similar goal were also being developed at about the same time [13].

Today, the Chimera Grid Tools package contains about 50 modules for pre-processing and post-processing of overset grid computations. This paper presents recent developments in three of the main modules in Chimera Grid Tools: OVERGRID, CGT Script Library, and OVERPLOT.

2. OVERGRID Graphical Interface

The OVERGRID graphical interface [14] was originally developed as a much-needed custom interface for performing structured overset grid generation. Over time, it has evolved into an interface for performing most pre-processing tasks before running a flow solver. The software is written primarily in C with some Fortran library calls.

Three-dimensional (3-D) graphics rendering is accomplished using OpenGL, while the user-interface widgets are created using Tcl/Tk [15]. The code has been ported to a variety of Unix, Linux, and Mac OS-X platforms.

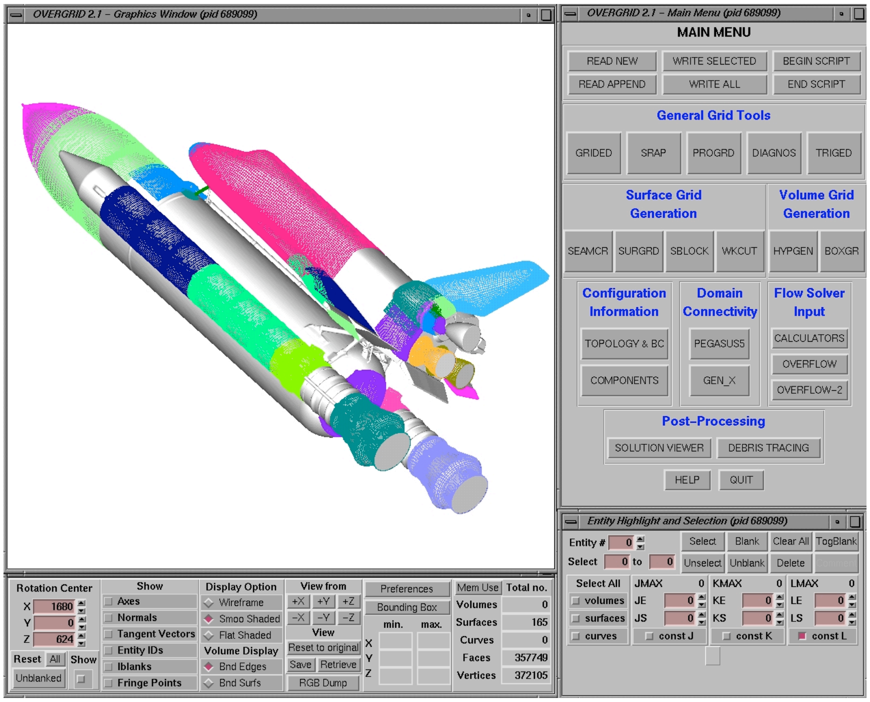

Figure 1. Main panels of OVERGRID graphical interface.

The main panels of OVERGRID consist of four windows as shown in Figure 1. The Graphics window (upper left) displays the 3-D geometry and grids. The Main Menu window (upper right) is used for file i/o and access to the various modules. The Controls window (lower left) is used to control various display attributes. The Selection window (lower right) is used for entity and subset highlight and selection.

OVERGRID allows structured grids and surface triangulations as input and output data types. Such data types can be treated as geometry definitions or the actual computational grids. Modules are available for

Grid editing – to swap or reverse grid indices, scale, translate, rotate or mirror grids, extract subsets, concatenate, extrapolate, split, revolve, smooth, or re-order grids.

Grid redistribution – to redistribute grid points on structured curve, surface or volume grids.

Grid projection – to project point, line, or surface subsets onto structured or triangulated surface definitions.

Grid diagnostics – to display grid attributes such as tangent and normal vectors, grid identity numbers, iblank locations, chimera fringe and orphan points information, grid quality functions; and to perform negative Jacobians reporting and surface grid topology detection.

Surface grid generation – to generate structured surface grids using hyperbolic or algebraic methods, and to create wake cut surface grids on wing-like geometries.

Volume grid generation – to generate structured hyperbolic or Cartesian volume grids.

Object X-ray map creation – to create object X-ray maps for performing hole cutting [16].

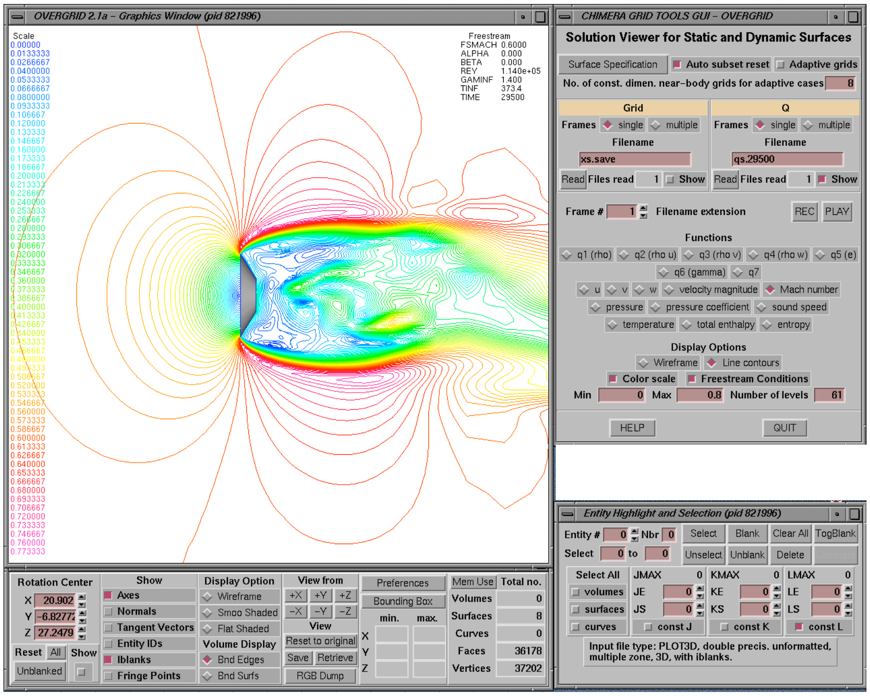

Figure 2. Solution viewer interface in OVERGRID showing color contours of Mach number .

Flow solver boundary conditions creation – to automatically detect and output boundary conditions for the OVERFLOW [17] and OVERFLOW-2 [18] flow solvers. Options are available to modify the boundary conditions and to view the boundary condition types by color.

The above features are described in more details in Ref. [14]. More recent developments include the following:

Calculators – to compute freestream properties such as Mach number, Reynolds number, temperature, density, etc. from input altitude, speed and reference length; to compute mass properties such as total mass and moments of inertia from input closed surface triangulation and mass density.

Component dynamics interface – to prepare inputs and animate results for multi-body dynamics simulations using the OVERFLOW-2 [18] or CART3D [19] flow solvers. Component hierarchy, linkage to grids, prescribed or 6-degree-of-freedom (6-dof) dynamics input can be prepared using the interface. Files are written in XML format using the Geometry Manipulation Protocol (GMP) [20] for input to flow solvers mentioned above. Prescribed component dynamics, and 6-dof dynamics without aero-forces and moments can be animated prior to running the flow solver. Six-dof dynamics output files from flow solvers, with aero-forces and moments effects, can be animated after running the flow solver.

Flow solver input preparation – to prescribe commonly used inputs such as freestream conditions, turbulence models, numerical time and spacing differencing schemes for the OVERFLOW and OVERFLOW-2 compressible Navier-Stokes flow solvers.

Solution viewer – to provide a means to rapidly view time-varying flow solutions generated by OVERFLOW-2 with component dynamics. From entering the solution viewer window, four mouse clicks on appropriate widgets will start an animation of time-varying color contours or grid wireframes of dynamics solutions. Default surfaces from grid and solution files are automatically loaded for display for 3-D cases. For 2-D cases only, all grids are automatically loaded for display where the grid dimensions are allowed to change with time as in adaptive grid simulations. Functions available for display include commonly used scalar variables such as Mach number, velocity components and magnitude, density, pressure, sound speed, and entropy, as well as turbulence model dependent variables and species densities. The intent and focus of this interface was for quick viewing of dynamic solutions. For more advanced solution analyses, more sophisticated visualization packages such as Fieldview are still required.

Debris tracing interface – to provide a means to prepare inputs for debris tracing computations using NASA's Debris code. Inputs may include debris shape, dimensions, density, surface offset distance, initial positions and velocities. An option is available to execute the Debris code and view debris trajectories and impact points for simple cases.

3. Chimera Grid Tools Script Library

For grid generation on complex configurations using overset structured grids, it is highly desirable to encode the procedure in a script that can easily reproduce all the steps in the process [21]. Key parameters such as grid spacings, stretching ratios, and geometry dimensions, can then be modified and the entire grid system regenerated automatically. The CGT Script Library, developed in the Tcl scripting language, was originally designed to assist the user in configuration scripting, i.e., given the surface grids, procedures are available to automatically build volume grids, and create the input files needed for domain connectivity, flow solver and post-processing [22].

Recently, a suite of new procedures was developed for the CGT Script Library for building surface and volume grid generation scripts. Typical surface grid generation procedures for complex configurations such as the flowliner and turbopump inducer combination in Figure 3 can involve hundreds of steps. Each step usually requires executing an independent batch code for some grid operations such as extracting a grid line, concatenating different grid parts, etc. Coding the input files for hundreds of instances of such steps is very tedious and error-prone. The new procedures in the CGT Script Library allows the execution of each step with a one-line procedure call.

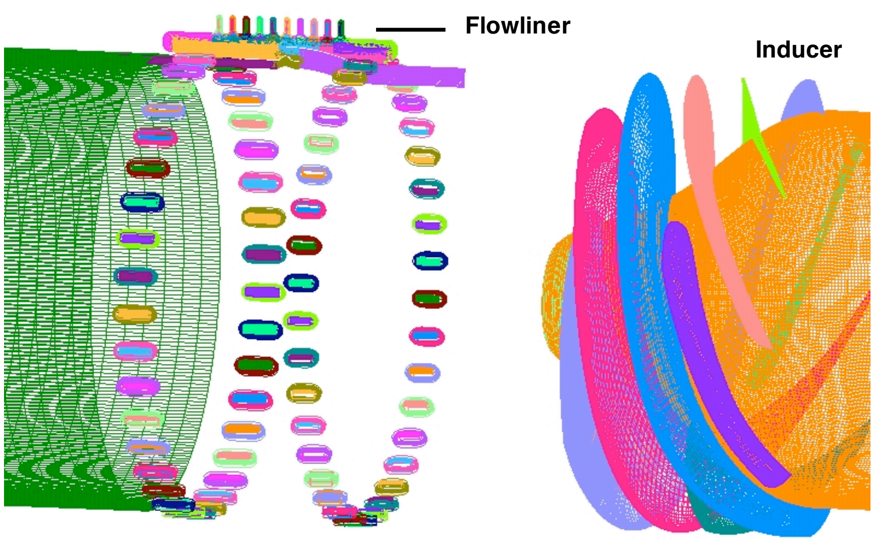

Figure 3. Overset grids for flowliner and turbopump inducer.

The new procedures in the library can be classified as follows:
1. File manipulation functions such as combining files and changing file formats.
2. Grid interrogation functions for returning information such as grid dimensions, coordinates, and arc lengths.
3. Grid editing, transformation, and redistribution functions that are analogous to similar functions in OVERGRID.
4. Grid generation functions for creating algebraic surface grids, hyperbolic and Cartesian volume grids.

5. Math functions.
6. Program execution and error checking functions.

By removing the need to write input files for each step, use of the new CGT Script Library procedures can typically result in a factor of 10 or more compact scripts, and about a factor of 3 or more faster script creation time.

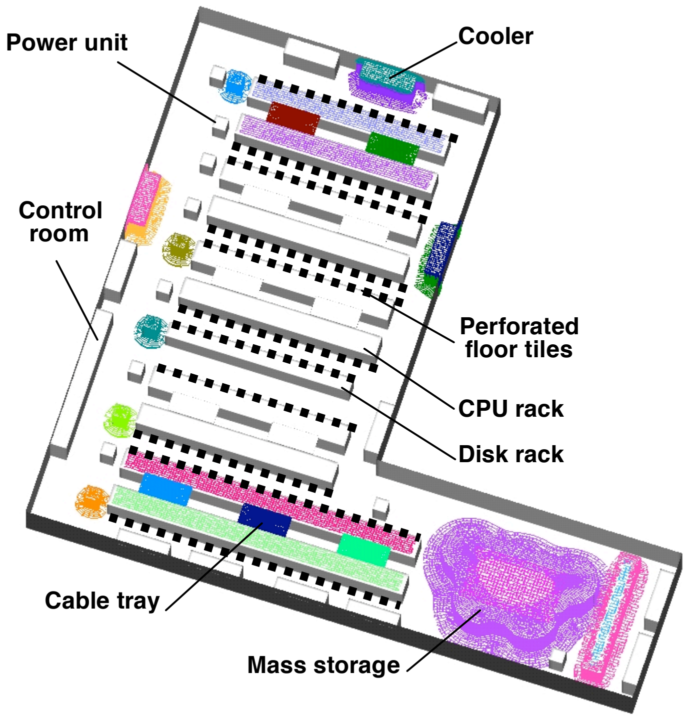

Figure 4. Surface grids for various components in computer room air flow simulation.

A recent application of the CGT Script Library is shown in Figure 4 for the air flow simulation in a modern computer room. Starting from blue-print drawings, the room and components geometry and grids were all created using a script utilizing the CGT Script Library in about 3 man-days. Components modeled in the room include the CPU and disk racks, cable trays, coolers, power units, and the mass storage system. The room dimensions, the component locations and dimensions, surface and volume grid attributes (grid spacings, stretching ratios), and the locations of specific boundary condition applications (e.g., cooler air in-take and floor-tile air injection) are all parameterized in the script. On changing any of the above parameters, the entire grid system and flow solver boundary conditions input can be regenerated in a few minutes on a desktop computer.

4. OVERPLOT Graphical Interface

For steady state computations, output from a flow solver typically includes histories of residuals of the flow equations for each grid, and the forces and moments acting on each component of the simulation. More sophisticated history output may also include the turbulence model equations residuals, the number of supersonic points and reverse flow points, and the minimum values of density and pressure in the entire flow field. For unsteady and moving-body problems, additional output may include residuals of Newton or dual time step sub-iterations, as well as dynamics data such as component center of mass coordinates and translational velocity, and component angular position and velocity. Some or all of the above history outputs can be used to determine convergence of a steady state computation, or to determine whether an unsteady computation has reached a quasi-periodic state or exhibits chaotic behavior.

The OVERPLOT graphical interface was designed to assist in the analysis of all the datasets mentioned above in one convenient environment. Acceptable history file formats are tuned to the output of the OVERFLOW and OVERFLOW-2 flow solvers, but simple conversion tools can be easily built to allow acceptance of output from other solvers. The graphical interface is written entirely in Tcl/Tk and hence is portable between Unix, Linux and Mac OS-X machines. All history data are displayed as 2-D plots using free-ware plotting packages such as GNUPLOT or XMGRACE. For each 2-D plot, OVERPLOT provides a convergence analysis tool where minimum, maximum, average and standard deviation values of the data and slope of the data are reported.

OVERPLOT provides 5 different analysis panels based on the type of the input history file:

1. Residual panel – residuals of flow equations or turbulence model equations for the main time steps and sub-iterations (Figure 5).
2. Force/moment panel – forces and moments of components with breakdown of contributions from pressure and viscous terms, and the mass flow rate through user-defined components (Figure 6).
3. Other flow parameters panel – minimum density/pressure, number of supersonic/reverse-flow points.
4. Dynamics panel – center of mass coordinates and velocity, angular positions and velocity of defined moving components.
5. Cp cuts panel – plots of multiple constant x, y, or z plane cuts on a component to display surface pressure variations.

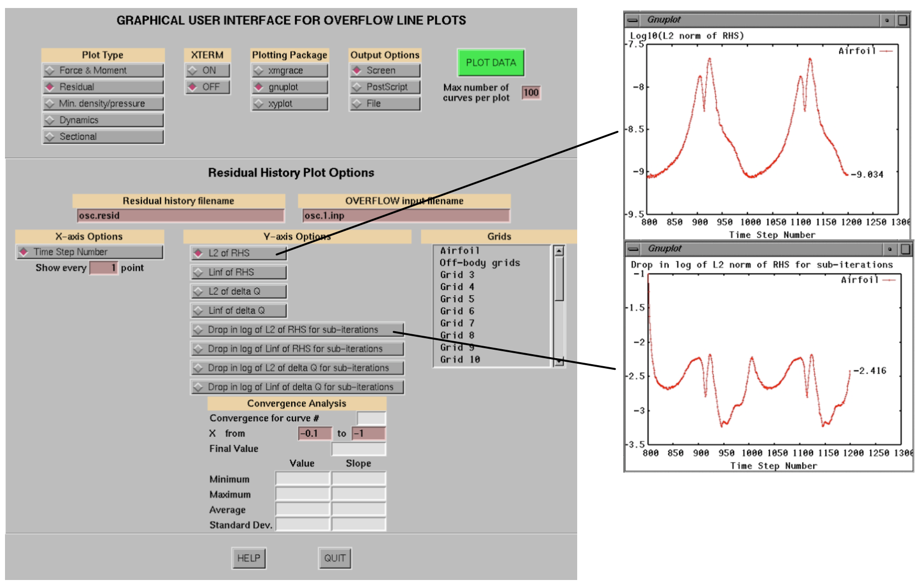

Figure 5. Residuals panel of OVERPLOT and two sample 2-D plots. The top plot shows the residual history of the main time step. The bottom plot shows the order of magnitude drop in residuals for the dual time step sub-iterations.

5. Concluding Remarks

Recent developments in three pieces of software for performing pre-processing and post-processing of overset grid computations have been presented. The OVERGRID interface has grown beyond performing grid generation tasks to performing various pre-processing tasks prior to running the flow solver. For multi-body dynamics simulations, OVERGRID can be used to animate and validate the dynamic motions of the components both prior to and after running the flow solver. The CGT Script Library was expanded to include many new procedures for performing different steps in surface and volume grid generation, resulting in an order of magnitude more compact scripts and much faster script creation time. The OVERPLOT interface was also extended to analyze sub-iteration residuals and history of dynamics data that arise as output in multi-body dynamics simulations. The software tools discussed here, together with domain connectivity software such as the PEGASUS 5 code [23] or the DCF module in OVERFLOW-2, and the OVERFLOW-2 flow solver, form a complete analysis package for steady flow simulations and unsteady multi-body dynamics flow simulations on complex configurations.

References

[1] Steger, J.L., Dougherty, F.C. and Benek, J.A., ``A Chimera Grid Scheme,'' *Advances in Grid Generation*, K.N. Ghia and U. Ghia, (Editors), ASME FED-Vol. 5, June, 1983

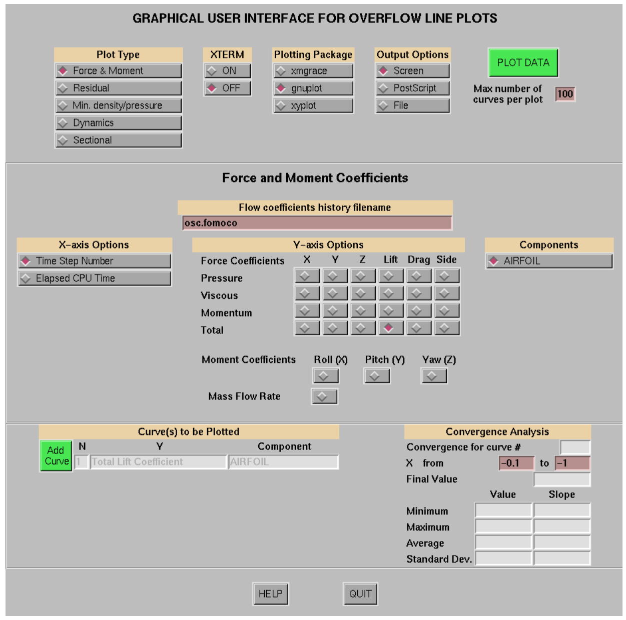

Figure 6. Forces and moments panel of OVERPLOT. List of defined components are automatically loaded under the Components column on the far right (only an AIRFOIL component is defined in this example).

[2] Meakin, R., ``Moving Body Overset Grid Methods for Complete Aircraft Tiltrotor Simulations,'' AIAA Paper 93-3350, *Proceedings of the 11th AIAA Computational Fluid Dynamics Conference*, Orlando, Florida, July, 1993.

[3] Slotnick, J.P., Kandula, M. and Buning, P.G., ``Navier-Stokes Simulation of the Space Shuttle Launch Vehicle Flight Transonic Flowfield Using a Large Scale Chimera Grid System,'' AIAA Paper 94-1860, *Proceedings of the 12th AIAA Applied Aerodynamics Conference*, Colorado Springs, Colorado, June, 1994.

[4] Atwood, C.A., ``Computation of a Controlled Store Separation from a Cavity,'' *J. of Aircraft*, Vol. 32, No. 4, pp. 846—852, 1995.

[5] Duque, E.P.N, Berry J.D., Budge A.M. and Dimanlig A.C.B., ``A Comparison of Computed and Experimental Flowfields of the RAH-66 Helicopter,'' *Proceedings of the 1995 American Helicopter Society Aeromechanics Specialist Meeting*, Fairfield County, Connecticut, 1995.

[6] Gee, K., Murman, S.M. and Schiff, L.B., ``Computation of F/A-18 Tail Buffet,'' *J. of Aircraft*, Vol. 33, No. 6, pp.~1181—1189, 1996.

[7] Srinivasan, G.R. and Klotz, S.P., ``Features of Cavity Flow and Acoustics of the Stratospheric Observatory For Infrared Astronomy,'' *Proceedings of the ASME Fluids Engineering Conference*, Vancouver, British Columbia, Canada, June, 1997.

[8] Meakin, R.L., ``Unsteady Aerodynamic Simulation of Static and Moving Bodies Using Scalable Computers,'' AIAA Paper 99-3302, *Proceedings of the 14th AIAA Computational Fluid Dynamics Conference*, Norfolk, Virginia, June, 1999.

[9] Gea, L.M., Halsey, N.D., Intemann, G.A. and Buning, P.G., ``Applications of the 3-D Navier-Stokes Code OVERFLOW for Analyzing Propulsion-Airframe Integration Related Issues on Subsonic

Transports,'' ICAS Paper 94-3.7.4, 19th Congress of the International Council of the Aeronautical Sciences, September, 1994.

[10] Wai, J., Herling, W.W. and Muilenburg, D.A., ``Analysis of a Joined-Wing Configuration,'' AIAA Paper 94-0657, January, 1994.

[11] Slotnick, J.P., An, M.Y., Mysko, S.J., Yeh, D.T., Rogers, S.E., Roth, K., Nash, S.M. and Baker, M.D., ``Navier-Stokes Analysis of a High Wing Transport High-Lift Configuration with Externally Blown Flaps,'' AIAA Paper 2000-4219, 18th AIAA Applied Aerodynamics Conference, Denver, Colorado, August, 2000.

[12] Chan, W.M., ``Hyperbolic Methods for Surface and Field Grid Generation,'' *CRC Handbook of Grid Generation*, Thompson, Soni and Weatherill, (Editors), CRC Press, 1998.

[13] Brown, D.L., Henshaw, W.D. and Quinlan, D.J.,``Overture: Object-Oriented Tools for Overset Grid Applications,'' AIAA Paper 99-3130, 17th AIAA Applied Aerodynamics Conference, Norfolk, Virginia, 1999.

[14] Chan, W.M., ``The OVERGRID Interface for Computational Simulations on Overset Grids,'' AIAA Paper 2002-3188, 32nd AIAA Fluid Dynamics Conference, St. Louis, Missouri, June, 2002.

[15] Welch, B., *Practical Programming in Tcl and Tk*, 3rd ed., Prentice Hall, 1999.

[16] Meakin, R.L., ``Object X-rays for Cutting Holes in Composite Overset Structured Grids,'' AIAA Paper 2001-2537, 15th AIAA Computational Fluid Dynamics Conference, Anaheim, California, June, 2001.

[17] Jespersen, D.C., Pulliam, T.H. and Buning, P.G., ``Recent Enhancements to OVERFLOW,'' AIAA Paper 1997-0644, 35th AIAA Aerospace Sciences Meeting, Reno, Nevada, January, 1997.

[18] Buning, P.G., Gomez, R.J. and Scallion, W.I., ``CFD Approaches for Simulation of Wing-Body Stage Separation,'' AIAA Paper 2004-4838, 22nd AIAA Applied Aerodynamics Conference, Providence, Rhode Island, August, 2004.

[19] Murman, S.M., Aftosmis, M.J., Berger, M.J., ``Simulations of 6-DOF Motion with a Cartesian Method,'' AIAA Paper 2003-1246, 41st AIAA Aerospace Sciences Meeting, Reno, Nevada, January, 2003.

[20] Murman, S.M., Chan, W.M., Aftosmis, M.J., Meakin, R.L., ``An Interface for Specifying Rigid-Body Motions for CFD Applications,'' AIAA Paper 2003-1237, 41st AIAA Aerospace Sciences Meeting, Reno, Nevada, January, 2003.

[21] Chan, W.M., Gomez, R.J., Rogers, S.E., and Buning, P.G., ``Best Practices in Overset Grid Generation,''AIAA Paper 2002-3191, 32nd AIAA Fluid Dynamics Conference, St. Louis, Missouri, June, 2002.

[22] Rogers, S.E., Roth, K., Nash, S.M., Baker, M.D., Slotnick, J.P., Whitlock, M., Cao, H.V., ``Advances in Overset CFD Processes Applied to Subsonic High-Lift Aircraft,'' AIAA Paper 2000-4216, 18th AIAA Applied Aerodynamics Conference, Denver, Colorado, August, 2000.

[23] Suhs, N.E., Rogers, S.E., and Dietz, W.E., ``PEGASUS 5: An Automated Pre-processor for Overset-Grid CFD,'' AIAA Paper 2002-3186, 32nd AIAA Fluid Dynamics Conference, St. Louis, Missouri, June, 2002.